DatView Documentation

1.0   General Information

1.1 Application Description

a) Name.   DatView

b)   Version Number,   1.0

c) Version Date,   October 2002

d)   Type.   Data viewer

e) Languages.   ANSI C. Uses Informix ESQL/C and , X11R6, Xt,
                          Motif 1.2 libraries

f)   Description.   DatView is a data viewer, displaying the data in both text and
                          graphic formats. It provides some capabilities to edit data,
                          print data and save data to flat files in different formats

1.2 Design Considerations

1.3 Assumptions application makes

linux system runnings X Windows with Motif and Informix properly installed and
configured

assumes Informix database using schema developed by the NWS National RFC
Archive Database Team

assumes system is using the apps_defaults token system similar to operational
AWIPS systems

assumes only raw data values can be edited, via a SHEF-encoded message

2.0   Configuration Information

a) apps_defaults tokens:
    datview_db_name: the database name
    datview_startdate: first date of data in database 'yyyy-mm-dd hh:mm:ss'
    datview_label_font: font used for label widgets

datview_text_font:	font used for text input widgets
datview_text2_font:	font used for main text data display widget
datview_list_font:	font used for list widgets
datview_bg_color:	background color for graphic data display
datview_fg_color:	foreground color for graphics data display
datview_ob_color1:	data display color 1
datview_ob_color2:	data display color 2 (used for forecast timeseries)
datview_ob_color3:	data display color 3 " " "
datview_ob_color4:	data display color 4 " " "
datview_ob_color5:	data display color 5 " " "
datview_ob_color6:	data display color 6 " " "
	After 6 forecasts are displayed colors rotate to color 1
datview_plot_font:	font used in graphic data display window
datview_plot_height:	height of graphic data display window (in pixels)
datview_plot_width:	width of graphic data display window (in pixels)
datview_save_dir:	directory to save flat files
datview_data_dir:	directory to store temporary shef files
	The above two tokens are optional, the default directory is the user's directory as defined by the HOME environment variable
datview_shef_dir:	directory where shefdecode expects to find data
datview_pil:	AFOS pil used for SHEF-encoding edited data
datview_office_header:	used for SHEF-encoding edited data (i.e. KTUA)

b) Environment variables
All enviroment variables needed to run INFORMIX must be configured properly
HOME   used for saving data if apps_defaults tokens not set
LOGNAME used in SHEF-encoded edited data file

3.0   User How-To (if its GUI, show images)

DatView consists of three windows: the main user interface, a data plotter and a text-based data viewer. There are also several other smaller windows that display on errors or other conditions. In the following description, please note that 'click' and 'select' imply a left-button mouse click, unless otherwise noted.

Figure 1 shows the main user interface. The user selects options on this interface to determine what data to display.

The user first chooses what data interval to view. This choice determines which table in the database gets queried. If the user wants to edit data, one of the Observed "Raw" or "Other" choices must be selected.

The next choice is the location. The user enters in the ID of the data location. If the user is unsure of the ID, but knows part of it or the name of the location, then they can choose the 'Find Ids' button to find the location ID. If the 'Find Ids' button is

Figure 1. Main GUI

Figure 2. 'Find Ids' pop-up

chosen, then a new window pops up (See Figure 2).   In this window, the user can user the popdown lists to filter the search by state, wfo or both. The choices can either be chosen from the popdown list or by typing in the appropriate text box.   The user then enters the search string and selects the 'Find' button. All locations whose ID or Name in the database that match the search criteria are then displayed in the lower portion of the window, in the format 'ID   Name". The user can then select an ID from the list and hit the 'Close' button. This will automatically put the selected 'ID' in the location text box. If no ID is selected, the location text box remains empty (or retains its original contents). Note that while the 'Find Ids' window is open, other windows are unavailable to use.

The user then selects the desired PE, Duration, Type-Source, Extremum and Probability from the respective popdown lists, or enters the values in the text box. The available options in these lists are read from the database shefpe, shefdur, shefts, shefex and shefprob tables. Note that 'All' is an option under these choices.

The next choices the user determines are the range of dates to look for data. The user can click on the appropriate value (year, month, day, etc) and type in the new value or click on the arrows to move the date and time forward or backward. The default starting date is determined from the apps_defaults token datview_startdate, and the default ending date is the current date + one day. (For forecast data, the retrieval method looks for the basistime within the date range).

The next two choices basically determine whether the data text and plot windows are displayed. The user can toggle these on or off depending on need.

The next set of toggles determine what data is displayed in the data text window. These can be toggled on or off and are all the available columns in the database tables. If a toggle is set, but the column does not exist in the current table, 'n/a' is displayed in the text window.

Once all the necessary information has been chosen, the 'Query' button is clicked to extract the data from the database.

The 'Exit' button quits the program upon selection.

Figure 3 show the data text display window. This window consists of a data view window with several options at the bottom.

The first option is 'Print'. If the user selects this button, the text, as displayed in the text window, is sent to the printer using the system 'lp' command.

The second option is 'Save'. If the user selects this button, the data is saved to either of three types of files, depending upon which button is toggled on in the 'Save Format' choice box. If 'ASCII Text' is chosen, the data is saved to a text file exactly as displayed. If 'Comma Delimited' is selected, the data is saved containing ALL the information from each row that the query matched, regardless of the toggles set in the main window.   Each data column is separated by a comma. If 'SHEF-encoded' is selected, a SHEF-encoded message is created. With either choice, a window pops up showing the current save directory (apps_defaults token datview_save_dir or the user's HOME directory). The user then enters the new filename or selects an existing filename. Note that no warnings are given if the user chooses an existing filename. That file will be overwritten.

```
                          Datview Text

                   BKLT2 Toledo Bend 12.5 NE

LID        PE Dur TS Obstime            Value
BKLT2      QT D    RZ 2002-10-23 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-22 13:00:00 0.78
BKLT2      QT D    RZ 2002-10-21 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-20 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-19 13:00:00 1.05
BKLT2      QT D    RZ 2002-10-18 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-17 13:00:00 0.77
BKLT2      QT D    RZ 2002-10-16 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-15 13:00:00 0.83
BKLT2      QT D    RZ 2002-10-14 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-13 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-12 13:00:00 1.03
BKLT2      QT D    RZ 2002-10-11 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-10 13:00:00 0.81
BKLT2      QT D    RZ 2002-10-09 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-08 13:00:00 0.81
BKLT2      QT D    RZ 2002-10-07 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-06 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-05 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-04 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-03 13:00:00 0.17
BKLT2      QT D    RZ 2002-10-01 13:00:00 2.44
BKLT2      QT D    RZ 2002-09-30 13:00:00 0.19
BKLT2      QT D    RZ 2002-09-29 13:00:00 0.20
BKLT2      QT D    RZ 2002-09-28 13:00:00 4.13
BKLT2      QT D    RZ 2002-09-27 13:00:00 4.68
BKLT2      QT D    RZ 2002-09-26 13:00:00 4.82
BKLT2      QT D    RZ 2002-09-25 13:00:00 4.72
BKLT2      QT D    RZ 2002-09-24 13:00:00 4.80
BKLT2      QT D    RZ 2002-09-23 13:00:00 0.20
BKLT2      QT D    RZ 2002-09-22 13:00:00 0.20
BKLT2      QT D    RZ 2002-09-21 13:00:00 4.11
BKLT2      QT D    RZ 2002-09-20 13:00:00 4.59
BKLT2      QT D    RZ 2002-09-19 13:00:00 4.70
BKLT2      QT D    RZ 2002-09-18 13:00:00 4.74
BKLT2      QT D    RZ 2002-09-17 13:00:00 4.76
BKLT2      QT D    RZ 2002-09-16 13:00:00 0.20
```

Print   Save   ┌─Save Format─────────────────────┐   Edit   Close
                 │ ◇ ASCII Text   ◇ SHEF Encoded │
                 │ ◇ Comma Delimited   │
                 └─────────────────────────────────┘

Figure 3. Text Window

The 'Edit' button is only activated when one of the first three observed data types is chosen and Show Missing values is on in the main user interface. Prior to selecting the 'Edit' button, the user must highlight part or all of the line or lines of data that are to be edited. To highlight, click and drag the cursor on the line or lines, or double or triple click on that line. Editing only allows the user to change the value, no other parts of the data can be edited (i.e. obstime, pe, etc.). If multiple lines are to be edited, they can only be set to that same value. Once the appropriate line or lines are highlighted, an edit window pops up (see Figure 4). The initial value is displayed (or the first value, if multiple lines are highlighted). The user then has three choices: set the value to missing, set the value to zero, or enter a new value. Once that has been done, the user

then clicks the 'Post' button. This creates a SHEF-encoded message and sends it to the SHEF-decoder to be parsed and entered in the database. The edit window then closes. If the user does not want to edit the data, then they can select the 'Close' button and no editing is done.   Note that the data does not change in the text or plot windows until the SHEF message has been posted to the database AND the query repeated.

The 'Close' button closes the text window. This has the same function turning the 'Text' toggle on the main interface off.


Figure 4. Edit window

Figure 5 shows the data plot window. This window plots the data values found from the query using the observation time or forecast validtime. The dates at the left and right of the window are determined by the data that is retrieved, not the starting and ending dates of the query. Observed data are plotted as 'o's, while forecast data is plotted as 'o's with lines connecting data from the same forecast time-series. The colors used in the display window are apps_defaults tokens. Forecast data is displayed with different colors to help differentiate the forecast time-series. The data window can show varying time frames with scrolling capability. The user can use the sliding bar at the bottom left to change the number of days displayed in the viewing window. The allowable values are determined by the range of the data's dates. The 'Close' button closes the data plot window. This functions similar to setting the 'Plot' toggle on the main interface to off.

Finally, several conditions can cause error windows to pop up (see Figure 6). In general, these windows occur if a needed item is left blank, as in a blank Location field on the main interface. These give a general error message and what was expected. Note that other windows become inactive until the 'OK' button is clicked.
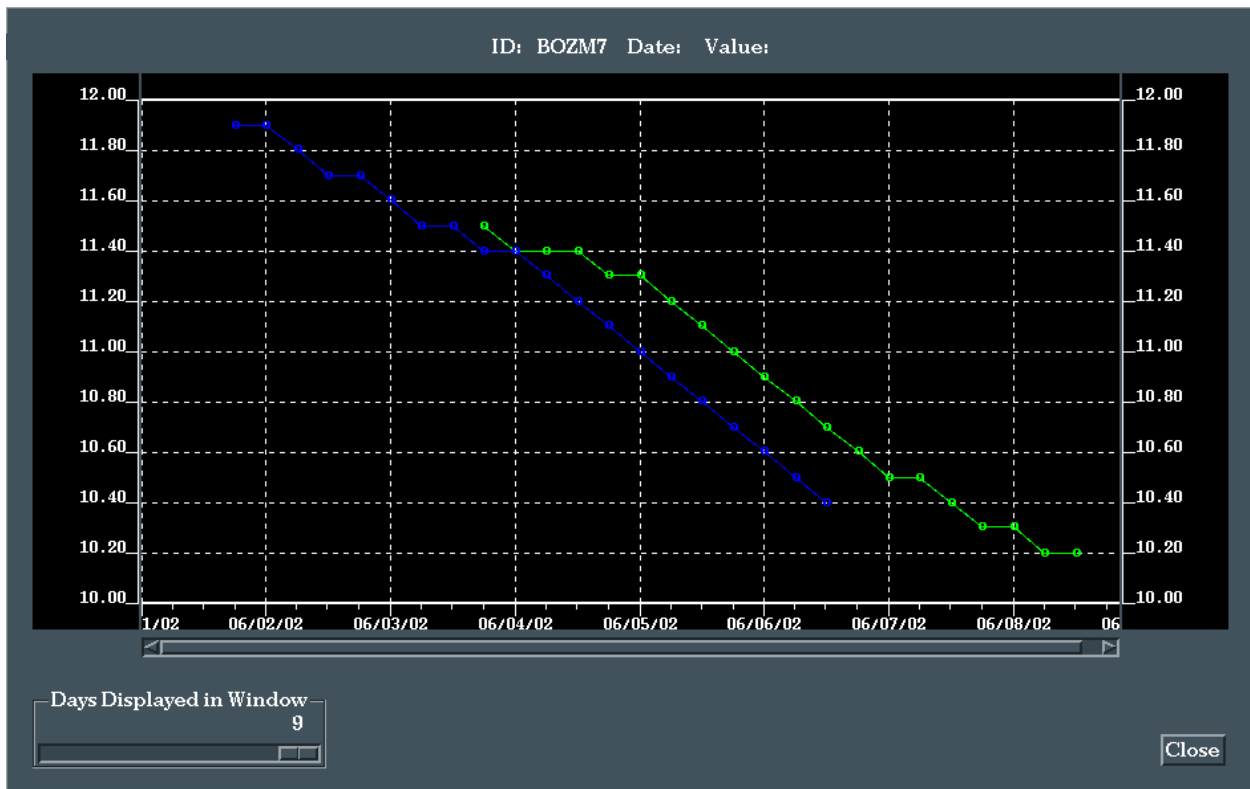
Figure 5. Data plot window
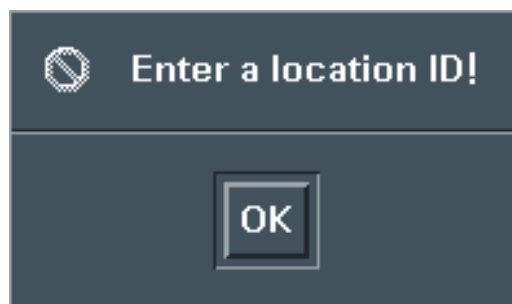
Figure 6. Typical error popup window

4.0   Troubleshooting Information

If run with a background window, error messages are displayed if INFORMIX related problems occur. All other problems, bugs should be reported to the maintenance programmer listed in section 6 of thsi documentation.

One known bug is that the font used in the edit window's text widgets is not the

correct font as defined by the app_defaults token for text fonts.

5.0   Installation Instructions

DatView is packaged as a tar file with all necessary directory structures, source code and a compiled executable. To install, place tar file in desired directory and type:

tar -xvf datview.tar

This will create a datview directory with source and bin subdirectories. If the compiled executable in the bin directory will not run, run make in the source directory to create a new executable and move this to the bin directory. Unless the X11, Xt, Motif (Xm) and Informix libraries are installed in non-standard locations, the Makefile should not need to be modified. If the newly compiled executable stiil does not work, contact the maintenance programmer listed in section 6.

6.0   Maintenance Information

Original Programmer:
James Paul
Arkansas-Red Basin River Forecast Center (ABRFC)
918-832-4109

Maintenance Programmer:
James Paul
Arkansas-Red Basin River Forecast Center (ABRFC)
918-832-4109

7.0   References

allowable colors to be used for Apps_default tokens are normally listed in: /usr/lib/X11/rgb.txt

xlsfonts may be used to see the list of fonts available on the system for defining Apps_default font tokens. xfd may also be used to display font characters.

Refer to NWS Hydrology Handbook Number 1 for SHEF code.

8.0   Future

Some features to hopefully be included in future releases include:
- not needing the 'Show Missing' values on for editing
- using the duration code to only show those valid obstimes (i.e. not showing the 15 minute intervals for hourly data)
- crosshair tracking on the data plot
- allow display of pairedvalues table
- allow inserting of data
- allow editing/displaying of comment